

**IN THE CLAIMS:**

All pending claims and their present status are produced below.

1. (Previously Presented) In a multithreaded analytic application executed by a source computer system and capable of concurrent execution of multiple session threads, a method for transferring data, the method comprising:
  - receiving an incoming request for analytic data resident in a mass storage unit on the source computer system;
  - authenticating the incoming request;
  - spawning a session thread that reads and parses a command received via the incoming request, the command for sending the data to a second computer system; and
  - concurrently executing a plurality of data transformation threads within the session thread, comprising
    - a reader thread that reads data and writes at least a part of the data to a first data block buffer;
    - a compressor thread that compresses the part of the data in the first data block buffer into a compressed data block and writes the compressed data block to a second data block buffer;
    - an encryptor thread that encrypts the compressed data block in the second data block buffer into an encrypted and compressed data block and writes the encrypted and compressed data block to a third data block buffer;
    - and
    - a writer thread that reads the encrypted and compressed data block in the third data block buffer and sends the encrypted and compressed data block to the second computer.

2. (Previously Presented) The method of Claim 1 further comprising:  
verifying that data transfer to the second computer system is complete.
3. (Previously Presented) The method of Claim 1 further comprising:  
verifying that data transfer to the second computer system is without error.
4. (Previously Presented) The method of Claim 1 wherein the source computer system and the second computer system are networked via the Internet.
5. (Previously Presented) The method of Claim 1 wherein the data comprises data processed by an analytic application.
6. (Previously Presented) The method of Claim 1 wherein the incoming request uses Extensible Markup Language (XML).
7. (Previously Presented) The method of Claim 1 wherein spawning a session thread further comprises:  
translating the command into a plurality of tasks;  
storing the tasks in a task table in a given order; and  
executing the tasks in order until a task ending the session thread is found.
8. (Previously Presented) The method of Claim 1 wherein the first data block buffer and the second data block buffer are substantially equal in size and wherein enough compressed data blocks are accumulated to fill the second data block buffer before the compressor thread writes to a second data block buffer.

9. (Previously Presented) The method of Claim 1 wherein the second data block buffer and the third data block buffer are substantially equal in size and wherein enough encrypted and compressed data blocks are accumulated to fill the third data block buffer before the encryptor thread writes to the third data block buffer.

10. (Previously Presented) The method of Claim 1 further comprising:  
restoring a connection with the second computer system when an ongoing connection is lost; and  
resuming transfer of data to the second computer system at the point in the data where the ongoing connection was lost.

11. (Previously Presented) In a first multithreaded analytic application executed by a target computer system and capable of concurrent execution of multiple session threads, a method for receiving data transferred from a source computer, the method comprising:  
issuing a request for data to the source computer system on which the data resides, the source computer system executing a second multithreaded analytic application;  
spawning a session thread in response to a message from the source computer system;  
receiving from the source computer system at least one encrypted and compressed data block of the data; and  
concurrently executing a plurality of data transformation threads within the session thread, comprising  
a reader thread for writing the encrypted and compressed data block to a first data block buffer;  
a decryptor thread for decrypting the encrypted and compressed data block into a compressed data block and writing the compressed data block to a second data block buffer; and

a decompressor thread for decompressing the compressed data block in the second data block buffer and writing a resultant data block to a third data block buffer.

12. (Previously Presented) The method of Claim 11 further comprising: verifying that data transfer from the source computer system was complete.

13. (Previously Presented) The method of Claim 11 further comprising: verifying that data transfer from the source computer system was without error.

14. (Previously Presented) The method of Claim 11 wherein the target computer system and the source computer system are networked via the Internet.

15. (Previously Presented) The method of Claim 11 wherein the data comprises data processed by an analytic application.

16. (Previously Presented) The method of Claim 11 wherein a plurality of encrypted and compressed data blocks accumulate before the decryptor thread executes.

17. (Previously Presented) The method of Claim 11 wherein a plurality of compressed data blocks accumulate before the decompressor thread executes.

18. (Previously Presented) The method of Claim 11 further comprising: restoring a connection with the source computer system when an ongoing connection is lost; and

resuming transfer of data from the source computer system at the point in the data where the ongoing connection was lost.

19. (Previously Presented) A source computer system comprising:

a bus;

a memory unit coupled to the bus; a multithreaded analytic application stored in the memory unit, and comprising:

a listener object for receiving an incoming request for data resident in a mass storage unit on the source computer system;

protocol for authenticating the incoming request;

a session manager object for spawning a session thread that reads and parses a command received via the incoming request, the command for sending the data to a second computer system;

a reader channel object for reading data and writing at least a part of the data to a first data block buffer;

a compressor channel object for compressing the part of the data in the first data block buffer into a compressed data block and writing the compressed data block to a second data block buffer;

an encryptor channel object for encrypting the compressed data block in the second data block buffer into an encrypted and compressed data block and writing the encrypted and compressed data block to a third data block buffer; and

a writer channel object for reading the encrypted and compressed data block in the third data block buffer and sending the encrypted and compressed data block to the second computer, wherein the application executes the reader channel object, the compressor channel

object, the encryptor channel object, and the writer channel object concurrently; and  
a processor coupled to the bus, the processor configured for executing the multithreaded analytic application.

20. (Previously Presented) The source computer system of Claim 19 wherein the processor is further configured for verifying that data transfer to the second computer system is complete.

21. (Previously Presented) The source computer system of Claim 19 wherein the processor is further configured for verifying that data transfer to the second computer system is without error.

22. (Previously Presented) The source computer system of Claim 19 wherein the source computer system and the second computer system are networked via the Internet.

23. (Previously Presented) The source computer system of Claim 19 wherein the data comprises data processed by an analytic application.

24. (Previously Presented) The source computer system of Claim 19 wherein the incoming request uses Extensible Markup Language (XML).

25. (Previously Presented) The source computer system of Claim 19 wherein the session manager object is further configured for:

translating the command into a plurality of tasks;  
storing the tasks in a task table in a given order; and  
executing the tasks in order until a task ending the session thread is found.

26. (Previously Presented) The source computer system of Claim 19 wherein the first data block buffer and the second data block buffer are substantially equal in size and wherein the compressor channel object is further configured for:

accumulating compressed data blocks before data are written to the second data block buffer, wherein enough compressed data blocks are accumulated to fill the second data block buffer.

27. (Previously presented) The source computer system of Claim 19 wherein the second data block buffer and the third data block buffer are substantially equal in size and wherein the encryptor channel object is further configured for:

accumulating encrypted and compressed data blocks before data are written to the third data block buffer, wherein enough encrypted and compressed data blocks are accumulated to fill the third data block buffer.

28. (Previously Presented) The source computer system of Claim 19 wherein the processor is further configured for:

restoring a connection with the second computer system when an ongoing connection is lost; and  
resuming transfer of data to the second computer system at the point in the data where the ongoing connection was lost.

29. (Previously Presented) A target computer system comprising:

a bus;

a memory unit coupled to the bus;

a multithreaded analytic application stored in the memory unit, and comprising:

a first session thread for issuing a request for data to a source computer system on which the data resides;

a session manager object for spawning a session thread in response to a message from the source computer system;

a listener object for receiving from the source computer system at least one encrypted and compressed data;

a reader channel object for reading data and writing at least part of the encrypted and compressed data to a first data block buffer;

a decryptor channel object for decrypting the encrypted and compressed data block into a compressed data block and writing the compressed data block to a second data block buffer; and

a decompressor channel object for decompressing the compressed data block in said the second data block buffer and writing a resultant data block to a third data block buffer, wherein the application executes the reader channel object, the decryptor channel object, and the decompressor channel object concurrently; and

a processor coupled to the bus, the processor configured for executing the multithreaded analytic application.

30. (Previously presented) The target computer system of Claim 29 wherein the processor is further configured for verifying that data transfer from the source computer system was complete.

31. (Previously Presented) The target computer system of Claim 29 wherein the processor is further configured for verifying that data transfer from the source computer system was without error.

32. (Previously Presented) The target computer system of Claim 29 wherein the target computer system and the source computer system are networked via the Internet.



33. (Previously Presented) The target computer system of Claim 29 wherein the data comprises data processed by an analytic application.

34. (Previously Presented) The target computer system of Claim 29 wherein the decryptor channel object is further configured for accumulating encrypted and compressed data blocks before decrypting the encrypted and compressed data blocks.

35. (Previously Presented) The target computer system of Claim 29 wherein the decompressor channel object is further configured for  
accumulating compressed data blocks before decompressing the compressed data  
blocks.

36. (Previously Presented) The target computer system of Claim 29 wherein the processor is further configured for:  
restoring a connection with the source computer system when an ongoing connection  
is lost; and  
resuming transfer of data from the source computer system at the point in the data  
where the ongoing connection was lost.